REMARKS

This Preliminary Amendment is submitted to improve the form of the English translation as filed. It is respectfully requested that this Preliminary Amendment be entered in the above-referenced application.

In accordance with the foregoing, claims 1-15 have been canceled and claims 16-28 have been added. Thus, claims 16-28 are pending and are under consideration.

A substitute specification is also being filed herewith. The substitute specification is accompanied by a marked-up copy of the original specification. No new matter has been added.

If there are any questions regarding these matters, such questions can be addressed by telephone to the undersigned. Otherwise, an early action on the merits is respectfully solicited.

If any further fees are required in connection with the filing of this Preliminary Amendment, please charge same to our Deposit Account No. 19-3935.

Respectfully submitted,

STAAS & HALSEY LLP

By: Richard A. Gollhofer
Richard A. Gollhofer
Registration No. 31,106

700 Eleventh Street, N.W.
Suite 500
Washington, D.C. 20001
(202) 434-1500

Date: 8/15/01

SUBSTITUTE SPECIFICATION

TITLE OF THE INVENTION

METHOD AND SYSTEM FOR STORING AND ACCESSING AN OBJECT OF A COMPUTER

BACKGROUND OF THE INVENTION

1. Field of the Invention

[0001] The invention relates to a method and system for storing and accessing an object by a computer.

2. Description of the Related Art

[0002] Such a method and system are known from computer technology or from commercially available computers. They permit data to be stored on different storage media, for example in a main memory or on a hard disk. Nowadays a large number of types of memory are available, such as: RAM, ROM, hard disk, diskette, moving head disk, CDROM, etc.

[0003] When processing large quantities of data, for example during digital image processing, the general aim is to ensure rapid processing in order, despite the requirement for real-time capability with moving images, to be able to carry out a large number of calculations, for example, to ensure high image quality in the event of a transmission channel experiencing interference, or to make optimum use of a small bandwidth with images with as high a resolution as possible.

[0004] In such a context, an image is modified by what is referred to as a converter which transforms an image into a mapping determined by the values of the parameter as a function of, in most cases, a plurality of parameters. Such transformation/mapping requires a period of time which cannot be ignored, in particular when processing images. If an image which is modified according to the predefined definition is to be converted at every access operation, there is a significant loss of performance, which considerably adversely affects the efficiency of a system particularly in the case of image processing. This is also aggravated by the fact that in numerous applications a plurality of the abovementioned converters are connected in series, and accordingly a large number of mappings have to be carried out.

SUMMARY OF THE INVENTION

[0005] An objective of the invention is to provide a mechanism which permits a significant saving in time when accessing an object, and in which in particular the object is stored in an advantageous way.

[0006] To achieve this objective, a method for storing an object by a computer includes determining a second object by a first object being modified by a predefined process which has at least one parameter. An index is determined by reference to the at least one parameter. The second object is stored in a memory with referencing of the second object being carried out by reference to its index.

[0007] In one embodiment modification is carried out by a plurality of processes. Here, each of the plurality processes can have one or more parameters.

[0008] Preferably the index is determined as a uniquely defined index. The uniquely defined index permits an unmistakable identification of the stored second object.

[0009] It is to be noted here that referencing is understood to be referral to the stored object, preferably by the index. This referencing can be carried out by storing the index in conjunction with an entry address for the stored second object. This can be carried out, for example, by using a table, and when the index is accessed the entry address for the second object which is being looked for in the memory is obtained from the table. This entry address can be implemented as an offset or as a pointer to the storage location.

[0010] Alternatively, the referencing can be carried out by storing the index together with the second object in the memory. In this case, an access can take place by searching the memory for the index.

[0011] A combination of the two aforesaid referencing possibilities is also possible.

[0012] In addition, it is possible that before the second object is stored it is compressed. The compression advantageously results in a significant reduction in the storage space required to store the second object.

[0013] To achieve the objective, a method for accessing an object by means of a computer is also disclosed in which an index is determined from at least one parameter of a process. This

index is used to dereference a second object. If a stored second object can be found with respect to the index, this second object is accessed; if a second object cannot be determined with respect to the index, a predefined first object is determined by means of the process while taking into account its at least one parameter with respect to a new second object. The access is made to the newly determined second object.

[0014] One development consists in the fact that the new second object is stored in accordance with the method described above.

[0015] In particular, instead of the one process, a plurality of processes can be used, each of the plurality of processes having a predefinable number of parameters.

[0016] There is also an embodiment in which an already stored object can also be accessed if the at least one parameter is similar to the at least one parameter of the already stored second object within a predefined tolerance. This has the particular advantage that within this tolerance an already stored second object is accessed in all cases, and the second object does not need to be newly determined or calculated specially.

[0017] There is also a development in which the object contains information which can be displayed.

[0018] In particular, the object can be a digital image.

[0019] An additional embodiment consists in the fact that the process is a converter for modifying image data. In this case the at least one parameter of the process (of the converter here) is a specific variable for influencing this image data.

[0020] A significant advantage of the invention consists in the fact that the direct memory access to an object which has already been determined eliminates the laborious and time-consuming new calculation (by means of the at least one process). The direct access to the memory is in all cases quicker and the elimination of the new determination has positive effects on the performance and the resources of the system.

[0021] It is to be noted here that the aforesaid memory comprises, in particular, the customary types of memory: RAM, mass storage, hard disk, etc.

[0022] To achieve the objective, a system for storing an object by a computer is also disclosed in which a processor unit is provided which is configured in such a way that

a) a second object can be determined by a first object being modified by a predefined process which has at least one parameter;

b) an index can be determined by reference to the at least one parameter;

c) the second object is stored in a memory, referencing of the second object being carried out by reference to its index.

[0023] To achieve the objective, a system for accessing an object by a computer is also disclosed, which has a processor unit configured in such a way that

a) an index can be determined from at least one parameter of a process;

b) dereferencing of a second object takes place by reference to the index;

c) if a stored second object can be determined with respect to the index, this second object is accessed;

d) if a second object cannot be determined with respect to the index, a new second object is determined from a predefined first object by means of the process, and this newly determined second object is accessed.

[0024] These arrangements are in particular suitable for carrying out the method according to the invention or one of its developments explained above.

[0025] Exemplary embodiments of the invention are presented and explained below with reference to the drawing.

BRIEF DESCRIPTION OF THE DRAWINGS

[0026] In the drawings:

Fig. 1 is a block diagram of a method for storing an object according to the present invention;

Fig. 2 is a memory diagram showing a first referencing possibility;

Fig. 3 is a memory diagram showing a second reference possibility;

Fig. 4 is a flowchart illustrating an access to an object; and

Fig. 5 is a block diagram of a processor unit in a computer.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0027] Fig. 1 is a block diagram of a method for storing an object. A first object 101 is firstly subjected to conversion with the parameters p1, p2, p3 in a converter 104 in a processing block 103. Compression (cf. block 105) and determining of an index (cf. block 106) are then carried out. During the determining of an index a value (index) is uniquely determined from the first object and the method of conversion (number of processes with values of the respective parameters). To do this, an identification variable 108 is preferably determined from the first object which variable is also taken into account in the determining of the index. The identification variable 108 permits uniquely defined assignment or virtually uniquely defined assignment of the first object 101 to the identification variable is 108. Various methods of doing this are known, for example an assignment by an Internet link (URL address). The index determining means 106 supplies a uniquely defined index 107, the compression means 105 supplies a second object 102 on which the conversion 104 and compression 105 have been performed.

[0028] Fig. 2 and Fig. 3 each show a memory diagram of a referencing possibility. Fig. 2 contains a storage area 201 in which the second object (indicated here by the area 203) is stored. The index 202 (corresponds to the index 107 determined according to Fig. 1) is preferably located at the start of the stored second object 203. By reference to the index 202, it is possible to find the second object again in the storage area 201.

[0029] Another possibility is shown by Fig. 3. To access a storage area 301 quickly, a table 305 is provided which comprises an index field 303 and a pointer 302. The pointer 302 points directly to an address within the storage area 301. Instead of the pointing mechanism, an offset, which to the same extent differentiates, in the storage area 301, the entry address for the object associated with index 303,can form in the table entry 302. In Fig. 3, the second object is indicated by the area 304, and the pointer 302 points to the start of the second object 304. If the second object 304 is being looked for by reference to its index 303, all that is necessary is to search through the table 305 for the index, and the field of the pointer 302 which is associated with the index directly supplies the start address within the storage area 301 for the second object 304.

[0030] Fig. 4 is a flowchart which illustrates an access to an object. A first object 401 and a conversion function (a process) 402 with parameters p1, p2 and p3 are given. By reference to the first object, an identification variable 403 is determined, and a subindex TIx 404 is determined from the conversion function which can optionally include a plurality of converters. Both variables 403 and 404 together yield the index Ix. The index Ix is used to look for a second object which has already been stored in a storage area and which results from the first object after the process 402 has been carried out (cf. 405). If the search is successful, the found object is accessed (cf. block 407), and otherwise the second object is newly calculated and this newly calculated object is accessed (cf. block 408). The decision as to whether an access or a new calculation must take place is made in a block 406.

[0031] Fig. 5 is a block diagram of a processor unit PRZE. The processor unit PRZE comprises a processor CPU, a memory SPE and an input/output interface IOS which is used in different ways via an interface IFC: Via a graphic interface, output can be viewed on a monitor MON and/or is issued on a printer PRT. An entry is made via a mouse MAS or a keyboard TAST. The processor unit PRZE also has a data bus BUS, which ensures the connection of a memory MEM, of the processor CPU and of the input/output interface IOS. Furthermore, additional components, for example additional memory, data memory (hard disk) or scanner, can be connected to the data bus BUS.

SUBSTITUTE ABSTRACT

ABSTRACT OF DISCLOSURE

METHOD AND SYSTEM FOR STORING AND ACCESSING AN OBJECT BY A COMPUTER

A method for storing an object by a computer is disclosed, in which a second object is determined by a first object being modified by a predefined process which has at least one parameter. An index is determined by reference to the at least one parameter. The second object is stored in a memory, referencing of the second object being carried by reference to its index.

MARKED-UP COPY OF SUBSTITUTE SPECIFICATION

[Description]

TITLE OF THE INVENTION

METHOD AND [ARRANGEMENT] SYSTEM FOR STORING AND ACCESSING AN OBJECT [BY MEANS] OF A COMPUTER

BACKGROUND OF THE INVENTION

1. Field of the Invention

[0001] The invention relates to a method and [an arrangement] system for storing and accessing an object by [means of] a computer.

2. Description of the Related Art

[0002] Such a method and [arrangement] system are known from computer technology or from commercially available computers. They permit data to be stored on different storage media, for example in a main memory or on a hard disk. Nowadays a large number of types of memory are available, [a selection is] such as: RAM, ROM, hard disk, diskette, moving head disk, CDROM, etc.

[0003] When processing large quantities of data, for example during digital image processing, the general aim is to ensure rapid processing in order, despite the requirement for real-time capability with moving images, to be able to carry out a large number of calculations [in order], for example, to ensure high image quality in the event of a transmission channel experiencing interference, or [in order to be able] to make optimum use of a small bandwidth with images with as [a] high a resolution as possible.

[0004] In such a context, an image is modified by [means of] what is referred to as a converter which transforms an image into a mapping determined by the values of the parameter as a function of, in most cases, a plurality of parameters. Such transformation/mapping requires a period of time which cannot be ignored, in particular when processing images. If an image which is modified according to the predefined definition is to be converted at every access operation, there is a significant loss of performance, which considerably adversely affects the efficiency of a system particularly in the case of image processing. This is also aggravated by

the fact that in numerous applications a plurality of the abovementioned converters are connected in series, and accordingly a large number of mappings have to be carried out.

SUMMARY OF THE INVENTION

[0005] [The] An objective of the [inventive consists in disclosing] invention is to provide a mechanism which permits a significant saving in time when accessing an object, and in which in particular the object is stored in an advantageous way.

[0006] [This objective is achieved in accordance with the features of the independent patent claims. Developments of the invention can be found in the dependent claims]

[0007] [In order to] To achieve [the] this objective, a method for storing an object by [means of] a computer [is disclosed in which] includes determining a second object [is determined] by a first object being modified by [means of] a predefined process which has at least one parameter. An index is determined by reference to the at least one parameter. The second object is stored in a memory[,] with referencing of the second object being carried out by reference to its index.

[0008] [One] In one embodiment [consists in] modification [being] is carried out by [means of] a plurality of processes. Here, each of the plurality processes can have one or more parameters.

[0009] [One development consists in the fact that] Preferably the index is determined as a uniquely defined index. The uniquely defined index permits an unmistakable identification of the stored second object.

[0010] It is to be noted here that referencing is understood to be referral to the stored object, preferably by the index. This referencing can be carried out by storing the index in conjunction with an entry address for the stored second object. This can be carried out, for example, by [means of] using a table, and when the index is accessed the entry address for the second object which is being looked for in the memory is obtained from the table. This entry address can be implemented as an offset or as a pointer to the storage location.

[0011] Alternatively, the referencing can be carried out by storing the index together with the second object in the memory. In this case, an access can take place by searching the memory for the index.

[0012] A combination of the two aforesaid referencing possibilities is also possible.

[0013] In addition, [there is a development in which] it is possible that before the second object is stored it is compressed. The compression advantageously results in a significant reduction in the storage space required to store the second object.

[0014] [In order to] To achieve the objective, a method for accessing an object by means of a computer is also disclosed in which an index is determined from at least one parameter of a process. This index is used to dereference a second object. If a stored second object can be found with respect to the index, this second object is accessed; if a second object cannot be determined with respect to the index, a predefined first object is determined by means of the process while taking into account its at least one parameter with respect to a new second object. The access is made to the newly determined second object.

[0015] One development consists in the fact that the new second object is stored in accordance with the method described above.

[0016] In particular, instead of the one process, a plurality of processes can be used, each of the plurality of processes having a predefinable number of parameters.

[0017] There is also an embodiment in which an already stored object can also be accessed if the at least one parameter is similar to the at least one parameter of the already stored second object within a predefined tolerance. This has the particular advantage that within this tolerance an already stored second object is accessed in all cases, and the second object does not need to be newly determined or calculated specially.

[0018] There is also a development in which the object contains information which can be displayed.

[0019] In particular, the object can be a digital image.

[0020] An additional embodiment consists in the fact that the process is a converter for modifying image data. In this case the at least one parameter of the process (of the converter here) is a specific variable for influencing this image data.

[0021] A significant advantage of the invention consists in the fact that the direct memory access to an object which has already been determined eliminates the laborious and time-

consuming new calculation (by means of the at least one process). The direct access to the memory is in all cases quicker and the elimination of the new determination has positive effects on the performance and the resources of the system.

[0022] It is to be noted here that the aforesaid memory comprises, in particular, the customary types of memory: RAM, mass storage, hard disk, etc.

[0023] [In order to] To achieve the objective, [an arrangement] a system for storing an object by [means of] a computer is also disclosed in which [arrangement] a processor unit is provided which is configured in such a way that

a) a second object can be determined by a first object being modified by a predefined process which has at least one parameter;

b) an index can be determined by reference to the at least one parameter;

c) the second object is stored in a memory, referencing of the second object being carried out by reference to its index.

[0024] [In order to] To achieve the objective, [an arrangement] a system for accessing an object by [means of] a computer is also disclosed, which [arrangement] has a processor unit [which is] configured in such a way that

a) an index can be determined from at least one parameter of a process;

b) dereferencing of a second object takes place by reference to the index;

c) if a stored second object can be determined with respect to the index, this second object is accessed;

d) if a second object cannot be determined with respect to the index, a new second object is determined from a predefined first object by means of the process, and this newly determined second object is accessed.

[0025] These [arrangements] systems are in particular suitable for carrying out the method according to the invention or one of its developments explained above.

[0026] Exemplary embodiments of the invention are presented and explained below with reference to the drawing.

BRIEF DESCRIPTION OF THE DRAWINGS

[0027] In [said drawings] the drawings:

Fig. 1 [shows an outline with logic blocks] is a block diagram of a method for storing an object according to the present invention;

Fig. 2 [shows an outline with] is a memory diagram showing a first referencing possibility;

Fig. 3 [shows an outline with] is a memory diagram showing a second reference possibility;

Fig. 4 [shows an outline] is a flowchart illustrating an access to an object;

Fig. 5 [shows] is a block diagram of a processor unit [which can be used as] in a computer.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0028] Fig. 1 [illustrates an outline with logic blocks] is a block diagram of a method for storing an object. A first object 101 is firstly subjected to conversion with the parameters p1, p2, p3 in a converter 104 in a processing block 103. Compression (cf. block 105) and determining of an index (cf. block 106) are then carried out. During the determining of an index a value (index) is uniquely determined from the first object and the method of conversion (number of processes with values of the respective parameters). To do this, an identification variable 108 is preferably determined from the first object which variable is also taken into account in the determining of the index. The identification variable 108 permits uniquely defined assignment or virtually uniquely defined assignment of the first object 101 to the identification variable is 108. Various methods of doing this are known, for example an assignment by [means of] an Internet link (URL address). The index determining means 106 supplies a uniquely defined index 107, the compression means 105 supplies a second object 102 on which the conversion 104 and compression 105 have been performed.

[0029] Fig. 2 and Fig. 3 each show [an outline with] a memory diagram of a referencing possibility. Fig. 2 contains a storage area 201 in which the second object (indicated here by the area 203) is stored. The index 202 (corresponds to the index 107 determined according to Fig.

1) is preferably located at the start of the stored second object 203. By reference to the index 202, it is possible to find the second object again in the storage area 201.

[0030] Another possibility is shown by Fig. 3. [: in order to] To access a storage area 301 quickly, a table 305 is provided which comprises an index field 303 and a pointer 302. The pointer 302 points directly to an address within the storage area 301. Instead of the pointing mechanism, an offset, which to the same extent differentiates, in the storage area 301, the entry address for the object associated with index 303,can form in the table entry 302. In Fig. 3, the second object is indicated by the area 304, and the pointer 302 points to the start of the second object 304. If the second object 304 is being looked for by reference to its index 303, all that is necessary is to search through the table 305 for the index, and the field of the pointer 302 which is associated with the index directly supplies the start address within the storage area 301 for the second object 304.

[0031] Fig. 4 [shows an outline] is a flowchart which illustrates an access to an object. A first object 401 and a conversion function (a process) 402 with parameters p1, p2 and p3 are given. By reference to the first object, an identification variable 403 is determined, and a subindex TIx 404 is determined from the conversion function which can optionally include a plurality of converters. Both variables 403 and 404 together yield the index Ix. The index Ix is used to look for a second object which has already been stored in a storage area and which results from the first object after the process 402 has been carried out (cf. 405). If the search is successful, the found object is accessed (cf. block 407), and otherwise the second object is newly calculated and this newly calculated object is accessed (cf. block 408). The decision as to whether an access or a new calculation must take place is made in a block 406.

[0032] Fig. 5 [illustrates] is a block diagram of a processor unit PRZE. The processor unit PRZE comprises a processor CPU, a memory SPE and an input/output interface IOS which is used in different ways via an interface IFC: Via a graphic interface, output can be viewed on a monitor MON and/or is issued on a printer PRT. An entry is made via a mouse MAS or a keyboard TAST. The processor unit PRZE also has a [databus] data bus BUS, which ensures the connection of a memory MEM, of the processor CPU and of the input/output interface IOS. Furthermore, additional components, for example additional memory, data memory (hard disk) or scanner, can be connected to the [databus] data bus BUS.